This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1-50. (cancelled)

51 (new). A method of using a computer program, each instantiation of which comprises at least two modules, the method comprising:

loading a first module, there being a plurality of different contexts in which the computer program can be instantiated, said first module being loaded when the computer program is instantiated in any one of said plurality of contexts, said context being defined at least based on the identity of the user who is using the computer program;

identifying, from among said plurality of contexts, a first context that is presently in existence;

creating a label that describes said first context, wherein the content of the label that is created is different for each one of said plurality of contexts, said label comprising:

an identifier of said user;

using said label to identify a second module to be retrieved, wherein each different user who uses the computer program results in a different label to be generated and a different second module identified; and

loading said second module.

52 (new). The method of claim 51, wherein said label further comprises:

a name of a department of an organization in which the computer program is being used, wherein each different department in which the computer program is used results in a different label being generated and a different module identified.

53 (new). The method of claim 51, wherein said label further comprises:

a name of the computer program; and

a name of a subcomponent of the computer program.

54 (new). The method of claim 53, wherein said label further comprises:

a name of an organization in which the computer program is being used.

55 (new). The method of claim 54, wherein said label further comprises:

a name of a department of said organization in which the computer program is being used.

56 (new). The method of claim 55, wherein said first module includes instructions to signify the occurrence of events, and wherein said second module performs actions only in response to events signified by said first module.

57 (new). The method of claim 56, further comprising:

publishing an interface to said events and a format of said label to permit software developers to write modules that respond to said events and that are identified by a label in said format.

58 (new). A computer-readable medium having stored thereon information comprising:

a first module of a computer program, said first module being used with a plurality of instantiations of said computer program;

a plurality of second modules of said computer program, each of said modules corresponding to a different context in which said computer program can be instantiated, each of said different contexts in which said computer program can be instantiated causing said computer program to be used with a different one of said second modules, each of said second modules being indexed in a module store based on a label that is descriptive of the context to which said module corresponds, each of the labels comprising:

a name of a user; and

computer-executable instructions that cause said computer program to be loaded by performing acts comprising:

loading said first module; and

determining a current context, said current context comprising an

identity of a user of said computer program;

   creating a label that is descriptive of said context, said label comprising an identity of said user of said computer program;

   using said label to retrieve the one of said second modules that is identified by said label; and

   loading the retrieved second module, wherein each different user who uses said computer program causes a different label to be generated, and a different module retrieved for loading.

  59 (new). The computer-readable medium of claim 58, wherein said label further comprises:

   a name of a department of an organization in which the computer program is being used, wherein each different department in which the computer program is used results in a different label being generated and a different module identified.

  60 (new). The computer-readable medium of claim 58, wherein said label further comprises:

   a name of the computer program; and

   a name of a subcomponent of the computer program.

  61 (new). The computer-readable medium of claim 60, wherein said label further comprises:

   a name of an organization in which the computer program is being used.

  62 (new). The computer-readable medium of claim 61, wherein said label further comprises:

   a name of a department of said organization in which the computer program is being used.

63 (new). The computer-readable medium of claim 62, wherein said first module includes instructions to signify the occurrence of events, and wherein said second module performs actions only in response to events signified by said first module.

64 (new). The computer-readable medium of claim 63, further comprising:

publishing an interface to said events and a format of said label to permit software developers to write modules that respond to said events and that are identified by a label in said format.

65 (new). A system for operating a computer program that is customizable based on context, the system comprising:

a first module of the computer program, the first module being loadable with an instantiation of the computer program for each one of a plurality of different contexts in which the computer program can be used;

a plurality of second modules of the computer program, each of the plurality of second modules being identified by and associated with a module label, each of the labels corresponding to a different one of the plurality of contexts; and

software that instantiates the computer program by performing acts comprising:

loading said first module;

creating a label that identifies the context in which the computer program is being instantiated, the label comprising an identity of a user who will use the instantiation of the computer program;

selecting one of said second modules by comparing said label with the module labels of said second modules, such that any two different users of the computer program will cause different labels to be generated and different second modules selected;

retrieving the selected module from a module store in which said modules are indexed based on their respective labels; and

loading the retrieved second module.

66 (new). The system of claim 65, wherein said label further comprises:

a name of a department of an organization in which the computer program is being used, wherein each different department in which the computer program is used results in a different label being generated and a different second module selected.

67 (new). The system of claim 66, wherein said label further comprises:

a name of the computer program; and

a name of a subcomponent of the computer program.

68 (new). The system of claim 67, wherein said label further comprises:

a name of said organization.

69 (new). The system of claim 67, wherein said first module includes instructions to signify the occurrence of events, and wherein each of said second modules performs actions only in response to events signified by said first module.

70 (new). The system of claim 69, further comprising:

publishing an interface to said events and a format of said label to permit software developers to write modules that respond to said events and that are identified by a label in said format.